What Is Claimed Is:

1.   1.   A method for accessing objects stored outside of main memory in

2   an object-addressed memory hierarchy, comprising:

3   receiving a request to access an object, wherein the request includes an

4   object identifier for the object that is used to reference the object within the

5   object-addressed memory hierarchy;

6   using the object identifier to retrieve an object table entry associated with

7   the object;

8   examining a valid indicator within the object table entry;

9   if the valid indicator indicates the object is located in main memory, using

10   a physical address in the object table entry to access the object in main memory;

11   and

12   if the valid indicator indicates that the object is not located in main

13   memory, relocating the object into memory from a location outside of memory,

14   and then accessing the object in main memory.

1   2.   The method of claim 1, wherein the request to access the object is

2   received at a translator that translates between object identifiers (used to reference

3   objects in an object cache) and physical addresses (used to address objects in main

4   memory).

1   3.   The method of claim 2,

2   wherein prior to receiving the request at the translator, the request is

3   initially directed to the object cache;

4   wherein if the request causes a hit in the object cache, the object is

5   accessed in the object cache and the request is not sent to the translator; and

6   wherein if the request causes a miss in the object cache, the request is sent

7   to the translator.

1   4.   The method of claim 1, wherein relocating the object into main

2   memory involves using location information from the object table entry to

3   determine the location of the object outside of main memory.

1   5.   The method of claim 4, wherein the location information can

2   include:

3   a secondary storage address for the object;

4   a network address for the object;

5   a uniform (or universal) resource locator (URL) for the object;

6   another (possibly different) object identifier associated with the object; and

7   a physical address for a compressed block of objects containing the object.

1   6.   The method of claim 4, wherein the location information is

2   overloaded into a physical address field in the object table entry.

1   7.   The method of claim 1, wherein relocating the object into main

2   memory involves causing an object fault handler to execute in a central processing

3   unit (CPU) to relocate the object into main memory.

1   8.   The method of claim 1, wherein relocating the object into main

2   memory involves overlapping retrieval of multiple objects into main memory

3   from locations outside of main memory.

1   9.   The method of claim 1, wherein after relocating the object into

2   main memory, the method further comprises:

3   updating the valid indicator to specify that the object is located in main

4   memory; and

5   updating the physical address in the object table entry to specify the

6   location of the object in main memory.

1   10.   The method of claim 1, wherein the object is defined within an

2   object-oriented programming system.

1   11.   An apparatus that facilitates accessing objects stored outside of

2   main memory in an object-addressed memory hierarchy, comprising:

3   a receiving mechanism configured to receive a request to access an object,

4   wherein the request includes an object identifier for the object that is used to

5   reference the object within the object-addressed memory hierarchy;

6   a object table lookup mechanism configured to use the object identifier to

7   retrieve an object table entry associated with the object;

8   an access mechanism configured to,

9   examine a valid indicator within the object table entry,

10   if the valid indicator indicates the object is located in main

11   memory, to use a physical address in the object table entry to

12   access the object in main memory, and

13   if the valid indicator indicates that the object is not located

14   in main memory, to relocate the object into memory from a

15   location outside of memory, and to access the object in main

16   memory.

14

1   12.   The apparatus of claim 11, wherein the receiving mechanism is
2   contained within a translator that translates between object identifiers (used to
3   reference objects in an object cache) and physical addresses (used to address
4   objects in main memory).

1   13.   The apparatus of claim 12, further comprising the object cache,
2   wherein prior to receiving the request at the translator, the request is
3   initially directed to the object cache;
4   wherein if the request causes a hit in the object cache, the object is
5   accessed in the object cache and the request is not sent to the translator; and
6   wherein if the request causes a miss in the object cache, the request is sent
7   to the translator.

1   14.   The apparatus of claim 11, wherein while relocating the object into
2   main memory, the access mechanism is configured to use location information
3   from the object table entry to determine the location of the object outside of main
4   memory.

1   15.   The apparatus of claim 14, wherein the location information can
2   include:
3   a secondary storage address for the object;
4   a network address for the object;
5   a uniform (or universal) resource locator (URL) for the object;
6   another (possibly different) object identifier associated with the object; and
7   a physical address for a compressed block of objects containing the object.

15

1   16.   The apparatus of claim 14, wherein the location information is
2   overloaded into a physical address field in the object table entry.

1   17.   The apparatus of claim 11, wherein while relocating the object into
2   main memory, the access mechanism is configured to cause an object fault
3   handler to execute in a central processing unit (CPU) to relocate the object into
4   main memory.

1   18.   The apparatus of claim 11, wherein while relocating the object into
2   main memory the access mechanism is configured to overlap retrieval of multiple
3   objects into main memory from locations outside of main memory.

1   19.   The apparatus of claim 11, wherein after relocating the object into
2   main memory, the access mechanism is configured to:
3   update the valid indicator to specify that the object is located in main
4   memory; and to
5   update the physical address in the object table entry to specify the location
6   of the object in main memory.

1   20.   The apparatus of claim 11, wherein the object is defined within an
2   object-oriented programming system.

1   21.   A computer system that facilitates accessing objects stored outside
2   of main memory in an object-addressed memory hierarchy, comprising:
3   a processor;
4   a main memory;
5   the object-addressed memory hierarchy;

16

6   an object cache within the object-addressed memory hierarchy;

7   a translator that translates between object identifiers, used to address

8   objects in the object cache, and physical addresses, used to address objects in

9   main memory;

10   wherein the translator is configured to receive a request to access an object

11   after the request misses in the object cache, wherein the request includes an object

12   identifier for the object that is used to reference the object within the object-

13   addressed memory hierarchy;

14   a object table lookup mechanism with the translator configured to use the

15   object identifier to retrieve an object table entry associated with the object; and

16   an access mechanism configured to,

17   examine a valid indicator within the object table entry,

18   if the valid indicator indicates the object is located in main

19   memory, to use a physical address in the object table entry to

20   access the object in main memory, and

21   if the valid indicator indicates that the object is not located

22   in main memory, to relocate the object into memory from a

23   location outside of memory, and to access the object in main

24   memory.

17